Architectural Design

The code is separated into 2 packages:
1.Controller
2. Model

1. Controller: It has one class controller.java which interfaces with commandline interface. It accepts the command from CLI and delegates them to the functionality in model packages

2. Model : This contains all the data and the behaviour for the project. For this build this package contains 7 classes and 1 enum. The classes are as follows:
   - Continents: this file contains the structure of a continent and all the behaviours/methods applicable on a constructor object.
   - Country: this file contains the structure of a country and all the behaviours/methods applicable on a constructor object.
   - GamePlay: This calls different methods from other classes as required by the game.
   - Mapx: this file contains the structure of a map and all the behaviours/methods applicable on a constructor object.
   - Player: this file contains the structure of a Player and all the behaviours/methods applicable on a constructor object.
   - Database: It is a static class which implements singleton design pattern. It holds the list of continents and a list of players.
   - Graph: It is a file that defines the architecture of the game. The object of this class will hold the map in memory.
   - State: It has an Enum holds the list of constants that are used to hold the states and valid commands

Pictorial representation of the flow of the game is as follow:

Concordia

game start

initilize Game

Game phase/state → edil map

Game Command → load map → map Editor

phase change → Edit player

Edit Continent
Edit Cantry
Edit neighbor
validate map
Save map

Show map → gameplayer → populate Countries

All states

troop Armies

place Army → place All

reinforcement

reinforce

fortification

fortify → fortify none